ENERGYPLUS™

Guide for Interface Developers

Everything You Need to Know about EnergyPlus Input and Output

(to develop a user-friendly interface)

Date: April 6, 2004

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

Introduction

This document is intended to be usable by interface developers wishing to create user interfaces for EnergyPlus. It will give an overview of the essentials of the input-output structure of EnergyPlus and then will describe in some detail the parts of each. In addition, licensing issues and other pieces of the puzzle will be presented.

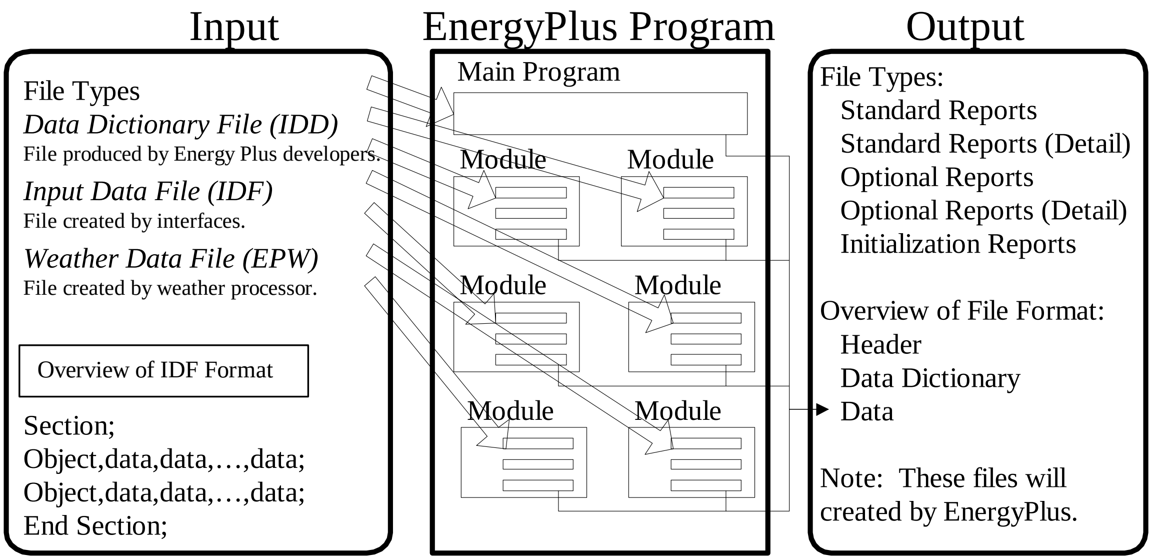

Figure 1. EnergyPlus Input/Output Overview

The diagram shown above should give the reader an overall picture of input-output in EnergyPlus. It can be seen as a linear process that includes the following steps:

1) The user enters building description (including internal space gains, HVAC arrangements, and Plant equipment properties) using the interface of their choice. In addition, the user specifies which non-default reports are desired and any optional variables from a predefined list of available simulation quantities.
2) The interface program writes the Input Data File (IDF) file, which includes the specification of any report items desired by the user.
3) EnergyPlus processes both the Input Data Dictionary (IDD) and the Input Data File (IDF) files with the "InputProcessor". The InputProcessor uses the specifications/rules defined in the IDD and interprets the IDF raw data. The InputProcessor is really quite "dumb" and only understands a few things about each field (alpha or numeric) qualified by certain key elements in the IDD (\ comments which are discussed later).
4) Each module in EnergyPlus has one or several routines (normally called "GetInput" routines) that obtain the information from the IDF file data. These subroutines decode the portion of the IDF file that is pertinent to the local module. These GetInput routines are more context sensitive than the InputProcessor and may perform further error detection. For example, the cooling coil module may read in the coil type and its associated parameters (number of rows, tube diameter, fin spacing, etc.).
5) EnergyPlus performs the simulation elements specified in the IDF. Output is generated as a continuous stream (for the most part) and must be interpreted into a more cohesive form by output processing. The user has control over which outputs are produced and when/how often.
6) EnergyPlus produces output as required by the user into one of the output files. These files can be readily processed into spreadsheet formats for graphing and other summarizing.

Interface Expectations

The input-output interfaces may be combined into a single program or may be available separately. The following attributes are expected from these interfaces.

Input Interface Attributes

The input interface agents will be expected to fulfill two main requirements:

- Ability to produce the input file for the simulation.
- Perform the consistency and value checks necessary to assure that the input file conforms to EnergyPlus requirements.

Additionally, the input interface agent might:

- Ability to warn users about potential output file size. It is expected that the data files generated by the EnergyPlus program will be significantly larger than the output files from its parent programs. As a result, users may be unaware that selecting too many reports could lead to enormous output files. It is recommended that some sort of checking be done to alert users when the term of the simulation and the number of reports selected eclipse some reasonable file size limit.
- Ability to perform parametric runs.

The method used by the input interface agent to accomplish these goals is left to the discretion of the interface developer.

Post-processing Interface Attributes

The post-processing agents will be expected to fulfill the main requirement:

- Ability to read all or selected output formats.

Additionally, a post-processing agent might:

- Ability to combine and summarize data (average, peak, total, etc.) and produce the various text and graphical reports requested by the user.
- Ability to handle multiple output files.

The method used by the post-processing agent to accomplish these goals is left to the discretion of the interface developer.

EnergyPlus Install Contents

EnergyPlus Interfaces will naturally need to access the installed EnergyPlus programs, library files, documentation. It will help to describe how EnergyPlus is installed (on a Windows™ computer). The EnergyPlus install is written using the WISE™ installation software.

The scheme of installing EnergyPlus includes a "root" directory/folder and all subsequent programs installed as part of the installation are child folders under the parent/root folder. Several optional components can be selected during install.

The basic (required) installation has crucial files installed in the parent folder – these include **EnergyPlus.exe**, **Energy+.idd** (the input data dictionary), **EPMacro.exe**. The basic installation also includes a child folder "DataSets" that contains the EnergyPlus "libraries". As distributed, EnergyPlus includes several library files, formatted in the standard EnergyPlus IDF format. These include thermal material properties, moisture material properties, glass and other properties for windows, constructions (material sandwiches which describe walls, windows, roofs), fluid properties, locations, design day definitions, and basic schedule definitions. There, of course, may be additional data sets added as well as future datasets edited for selectable use from the EPMacro program. The Templates folder is also included in the basic install. Currently, the Templates folder contains documentation and the HVAC Template files. These files can be used somewhat "automatically" to produce HVAC loop structures for running with EnergyPlus.

Optional components of the EnergyPlus install: **Documentation**, **EP-Launch**, **IDFEditor**, **Sample Files**, **Weather Converter**, **BLAST Translator**, **DOE-2 Translator**, and the IFC-to-IDF element. Those shown in bold are default – "selected" and must be un-selected during install by the user. A brief description:

Documentation: The EnergyPlus package includes a comprehensive set of documents intended to help the user and others understand the EnergyPlus program, usage, and other appropriate information. All documents are created in PDF (Portable Document Format). A shortcut to the Documentation folder and the "main-menu" document is included. The main-menu document is a navigation aid to the remainder of the documents. We may want to make the documentation a non-optional component.

EP-Launch: The components of EP-Launch are installed in the parent directory (help files installed in the documentation directory). Because the basic EnergyPlus program runs as a console application, many beta users did not understand how to make the program execute. While the developers may be able to tailor the EnergyPlus executable to be more callable under the Windows™ platform, this is still a useful program. EP-Launch uses the EPL-Run.bat file and prepends several "set" commands that are used in the bat file. It creates the actual batch file for the run as "RunEP.bat" and then calls the operating system to execute the file. Having the EPL-Run.bat file as external to the EP-Launch program means that others may tailor the batch file more appropriately to how things are run though this may not be preserved with a future EnergyPlus install.

IDFEditor: The IDFEditor is the simple editor that is distributed with EnergyPlus. As an interface, it is crude. However, it gets the job done. It uses the IDD and then reads and/or creates an IDF file. The objects are shown in the groups (see \group discussion below) and, when an existing file is used, will display how many of an object is found in the IDF. This program is installed in the Parent \ PreProcess \ IDFEditor folder.

Sample Files: The sample files include several IDF files along with the files the installed version of EnergyPlus created using these files. There are several possible child folders here, including the Misc child folder that will contain all the development sample files – but without having been run for the install. These files are installed in the Parent \ ExampleFiles folder and any appropriate child folders under that.

Weather Converter: The WeatherConverter program can process raw weather data in several formats into the EnergyPlus weather data format (epw). In addition, the

WeatherConverter program can be used to generate a simple report of the weather data as well as produce a .csv file of the format. The .csv (comma separated variable) format can easily be imported into spread sheet programs or other tables. This program is installed in the Parent \ PreProcess \ WeatherConverter folder.

BLAST Translator: The BLAST translator program can be used to convert a BLAST input file into a format that can be executed from EnergyPlus. Extensive system translation is not done with this program – mostly geometry and other space gain elements as well as zone oriented (i.e. People) schedules. If the BLAST input file contains thermostatic controls in the zones, then the EnergyPlus IDF file will include a purchased-air solution of that BLAST input file. Future versions of the BLAST translator may make use of the HVAC templates but no guarantee. This program is installed in the Parent \ PreProcess \ BLASTTranslator folder.

DOE-2 Translator: The DOE-2 translator program is similar to the BLAST Translator program but for DOE-2 files. DOE-2 translator output must be processed by the EPMacro program prior to running in EnergyPlus (the EP-Launch program/EPL-Run procedure does this automatically for all ".imf" files). This program is installed in the Parent \ PreProcess \ DOE-2Translator folder.

IFC-to-IDF Translator: The IFC-to-IDF translator set of programs includes the facility for using an IFC (Industry Foundation Class) file and creating an IDF file. This program is installed in the Parent \ PreProcess \ IFCtoIDF folder.

Input Overview

The general structure of the input files for EnergyPlus is plain text. Fields are comma delimited and each “line” is terminated with a semicolon. This allows for a very rudimentary input processor that can be instantly flexible to developer’s needs. However, it puts more burdens on the EnergyPlus developers to process the input information, supply defaults as needed, and perform validity checks. Guidelines were established for the input:

- Input will be a flat ASCII file with comma-delimited columns and each “line” (where each line can run over several physical file records) terminated with a semicolon.
- Input should be “readable”, “editable”, “simply parsed with few value checks or consistency checks”.
- Input, to the extent possible, should be easily maintainable and extendable.
- Input will be “object based”.
- Definitions in a data dictionary will define the input. The data dictionary should be self-documenting.
- All input units will be metric (SI). Conversions from “user units” will be done in the interface agents.

Two input files are necessary for the input processing. The first is the “data dictionary” which will specify the requirements for each item. The EnergyPlus Input Processor uses these requirements to process the “input data file” and report any anomalies found. Both input files have similar structures: 1) Sections – single lines/commands, which may help group the simulation input for readability and 2) Classes/Objects – data attributes for the simulation. Classes are the term used in the data dictionary – each class will specify the kind of data (alpha or numeric) that will be included in the simulation input. Objects are instances of these classes and appear in the IDF with appropriate data.

General Input Rules

The following rules apply to both the Input Data Dictionary and the Input Data File.

- The initial line of a definition or input MUST have a comma or semicolon.
- Fields do not extend over line boundaries. Usually, if a comma or (as appropriate) semi-colon is not the last field value on a line, one will be inserted. Of course, several fields may appear on a single line as long as they are comma separated.
- Commas delimit fields – therefore, no fields can have embedded commas. No error will occur but you won't get what you want.
- Blank lines are allowed.
- The comment character is a exclamation “!”. Anything on a line after the exclamation is ignored.
- Input records can be up to 500 characters in length. If you go over that, no error will occur but you won't get what you want.
- Each Section and Class/Object keyword can be up to 60 characters in length. Embedded spaces are allowed, but are significant (if you have 2 spaces in the section keyword – you must have 2 when you write the object keyword).
- Each Alpha string (including Section and Class/Object keywords) is mapped to UPPER case during processing. Get routines from the EnergyPlus code that use the Section and Object keywords automatically map to UPPER case for finding the item. The primary drawback with this is that error messages coming out of the input processor will be in UPPER case and may not appear exactly as input.
- Special characters, such as tabs, should NOT be included in the file. However, tabs can be accommodated and are turned into spaces.

Input Data Dictionary

The input data dictionary specifies the “definitions” for each line that will be processed in the input data file.

Structure in the input data dictionary allows for descriptions that may be useful for interface developers. The Input Processor ignores everything but the essentials for getting the “right stuff” into the program. Developers have been (and will continue to be) encouraged to include comments and other documentation in the IDD.

Internal to the data dictionary (using special “comment” characters) is a structured set of conventions for including information on each object. This is shown in section on Input Details below.

Rules specific to the Input Data Dictionary

In addition to the rules for both files (listed above), the IDD also has the limitation:

- Duplicate Section names and Duplicate Class names are not allowed. That is, the first class of an item named X will be the one used during processing. Error messages will appear if you try to duplicate definitions.

Input Data File

This is the only file that EnergyPlus uses to create the building simulation. The input is order-independent; data can appear in any order and will be retrieved and sorted as necessary by the EnergyPlus simulation modules. In addition, EnergyPlus allocates everything dynamically, so there are no limitations as to number of zones, surfaces, etc.

All numbers can be flexibly input and are processed into single precision variables (i.e. 1.0, 1.000, 1, .1E+1 are all processed equally).

Rules specific to Input Data file:

- Each Alpha string in the input data file can be up to 60 characters in length. Anything beyond that is truncated.

Input Details

This document does not cover the input "classes" in detail. For details on each class and examples of both input and output resulting from that class/object, please view the Input Output Reference document. In this document, we will show the "conventions" used in the IDD and provide limited examples for illustration purposes.

An intelligent editor (IDFEditor) has been written and can be used as an illustration of how the comments in the IDD might be used by Interface Developers to guide their developments. IDFEditor is described in the Getting Started document.

A full example of a very simple IDF is included in Appendix A to this document.

IDD Conventions

The following is a basic description of the structure of the IDD (it's actually taken directly from the IDD file). As noted within, ! signifies a comment character as does the \. \ has also been adopted as a convention for including more specific comments about each field in an object. These have been used with success in the IDFEditor and it is hoped the flexibility will provide other interface developers with useful information. Not all fields are filled in as of yet in the existing IDD – that will come over time.

```
! **************************************************************************
!
! Object Description
! ------------------
! To define an object (a record with data), develop a key word that is unique
! Each data item to the object can be A (Alphanumeric string) or N (numeric)
! Number each A and N.  This will show how the data items will be put into the
! arrays that are passed to the Input Processor "Get" (GetObjectItem) routines.
! All alpha fields are limited to 60 characters.  Numeric fields should be
! valid numerics (can include such as 1.0E+05) and are placed into single
! precision variables.
!
! NOTE: Even though a field may be optional, a comma representing that field
!       must be included (unless it is the last field in the object).  Since
!       the entire input is "field-oriented" and not "keyword-oriented", the
!       EnergyPlus Input Processor must have some representation (even if
!       blank) for each field.
!
! Object Documentation
! --------------------
! In addition, the following special comments appear one per line and
! are followed by a value.  Comments may apply to a field, and object
! or a group of objects.
!
! Field-level comments:
!
!  \field           Name of field
!                     (should be succinct and readable, blanks are encouraged)
!
!  \note            Note describing the field and its valid values
!
!  \required-field  To flag fields which may not be left blank
!                     (this comment has no "value")
!
!  \units           Units (must be from EnergyPlus standard units list)
!                   EnergyPlus units are standard SI units
!
!  \ip-units        IP-Units (for use by input processors with IP units)
!
```

```
!  \minimum          Minimum that includes the following value
!
!  \minimum>         Minimum that must be > than the following value
!
!  \maximum          Maximum that includes the following value
!
!  \maximum<         Maximum that must be < than the following value
!
!  \default          Default for the field (if N/A then omit entire line)
!
!  \autosizeable     Flag to indicate that this field can be used with the
!                    Auto Sizing routines/function to produce calculated results
!                    for the field.
!
!  \type             Type of data for the field -
!                      integer
!                      real
!                      alpha    (arbitrary string),
!                      choice   (alpha with specific list of choices, see \key)
!                      object-list (link to a list of objects defined elsewhere,
!                                   see \object-list and \reference)
!
!  \key              Possible value for "\type choice" (blanks are significant)
!                      use multiple \key lines to indicate all valid choices
!
!  \object-list      Name of a list of object names that are valid entries
!        for this field (used with "\reference")
!        see ZONE and Surface:HeatTransfer objects below for examples
!        ** Note that a field may have only one \object-list reference.
!        ** If it must reference more than one object-list, then a composite
!        object-list should be created which contains all valid references.
!
!  \reference        Name of a list of object names to which this object belongs
!        used with "\type object-list" and with "\object-list"
!        see ZONE and Surface:HeatTransfer objects below for examples:
!
!                        ZONE,
!                          A1 , \field Zone Name
!                               \type alpha
!                               \reference ZoneNames
!                        Surface:HeatTransfer,
!                          A4 , \field InsideFaceEnvironment
!                               \note Zone the surface is a part of
!                               \type object-list
!                               \object-list ZoneNames
!
!         For each zone, the field "Zone Name" may be referenced
!         by other objects, such as Surface:HeatTransfer, so it is commented
!         with "\reference ZoneNames"
!         Fields that reference a zone name, such as a Surface:HeatTransfer's
!         "InsideFaceEnvironment", are commented as
!         "\type object-list" and "\object-list ZoneNames"
!         ** Note that a field may have multiple \reference commands.
!         ** This is useful if the object belongs to a small specific
!         object-list as well as a larger more general object-list.
!
! Object-level comments:
!
!  \memo             Memo describing the object
!
!  \unique-object    To flag objects which should appear only once in an idf
!                      (this comment has no "value")
!
```

```
!  \required-object To flag objects which are required in every idf
!                     (this comment has no "value")
!
!  \min-fields      Minimum number of fields that should be included in the
!                   object.  If appropriate, the Input Processor will fill
!                   any missing fields with defaults (for numeric fields).
!                   It will also supply that number of fields to the "get"
!                   routines using blanks for alpha fields (note -- blanks
!                   may not be allowable for some alpha fields).
!
! Group-level comments:
!
!  \group          Name for a group of related objects
!
!
! Notes on comments
! -----------------
!
! 1.  If a particular comment is not applicable (such as units, or default)
! then simply omit the comment rather than indicating N/A.
!
! 2.  Memos and notes should be brief (recommend 5 lines or less per block).
! More extensive explanations are expected to be in the user documentation
!
! 3.  19Jul01 - Surface name object lists have been revised as follows:
!
!  SurfaceNames
!    contains the following objects:
!      Surface:HeatTransfer
!    referenced by the following object fields:
!      MovableInsulation --> SurfaceName
!      Surface:HeatTransfer:Sub --> Base Surface Name
!      Surface:Shading:Attached --> Base Surface Name
!      LOW TEMP RADIANT SYSTEM:HYDRONIC --> Surface name
!                           (name of surface system is embedded in)
!      LOW TEMP RADIANT SYSTEM:ELECTRIC --> Surface name
!                           (name of surface system is embedded in)
!
!  SurfAndSubSurfNames
!    contains the following objects:
!      Surface:HeatTransfer
!      Surface:HeatTransfer:Sub
!    referenced by the following object fields:
!      COMIS Surface Data --> Surface Name
!
!  AllHeatTranSurfNames
!    contains the following objects:
!      Surface:HeatTransfer
!      Surface:HeatTransfer:Sub
!      Surface:HeatTransfer:InternalMass
!    referenced by the following object fields:
!      PEOPLE --> Surface Name
!      HIGH TEMP RADIANT SYSTEM --> surface to which radiant energy
!                                 from heater is distributed 1-20
!
!  OutFaceEnvNames
!    contains the following objects:
!      Surface:HeatTransfer
!      OtherSideCoefficients
!      Surface:HeatTransfer:Sub
!    referenced by the following object fields:
!      Surface:HeatTransfer --> OutsideFaceEnvironment Object
!      Surface:HeatTransfer:Sub --> OutsideFaceEnvironment Object
```

```
!
! 4.  19Jul01 - COMIS opening name groups have been revised as follows:
!  AirFlowNames
!    contains the following objects:
!      COMIS AIR FLOW:Crack
!      COMIS AIR FLOW:Opening
!    referenced by the following object fields:
!      COMIS Surface Data --> Air Flow Crack or Opening Type
!
! **********************************************************************
```

The LOCATION object will serve as an example.

```
Location,
       \unique-object
       \min-fields 5
  A1 , \field LocationName
       \required-field
       \type  alpha
  N1 , \field Latitude
       \units deg
       \minimum -90.0
       \maximum +90.0
       \default 0.0
       \note + is North, - is South, degree minutes represented in decimal (i.e. 30 minutes is .5)
       \type real
  N2 , \field Longitude
       \units deg
       \minimum -180.0
       \maximum +180.0
       \default 0.0
       \note - is West, + is East, degree minutes represented in decimal (i.e. 30 minutes is .5)
       \type real
  N3 , \field TimeZone
       \units hr (decimal)
       \minimum -12.0
       \maximum +12.0
       \default 0.0
       \note  Time relative to GMT.
       \type real
  N4 ; \field Elevation
       \units m
       \minimum -300.0
       \maximum< 6096.0
       \default 0.0
       \type real
```

First, the object name is given. (Location) This is followed by a comma in both the definition (IDD) and in an input file (IDF). In fact, all fields except the terminating field of an IDD class object and IDF object are followed by commas. The final field in an IDD class object or in an IDF object is terminated by a semi-colon.

Next is an alpha field, the location name. As noted above, for input, spaces are significant in this field. The main inputs for Location are numeric fields. These are numbered (as is the alpha field) for convenience. The \ designations will show various information about the objects as described above in the IDD conventions discussion. Of importance for reading this document are the units and possible minimum and maximum values for a field. Defaults are produced if applicable and if the field is blank in the IDF. For this example, that won't work because there is no default AND the Location name is a required field.

The \minimum, \maximum and \default comments are automatically processed in the Inptu Processor for numeric fields. Any infractions of the \minimum, \maximum fields will be automatically detected and messages will appear in the standard error file. After all the input is checked, infractions will cause program termination (before the bulk of the simulation is completed). Defaults are also enforced if you leave the numeric field blank.

\min-fields deserves some explanation. This has object level enforcement. When min-fields is specified for an object, it has meaning to the InputProcessor – telling the IP the number of fields that must be returned to a GetInput routine regardless of how many fields may actually appear in the IDF. So, min-fields is actually an automatic assistance in most instances.

Some objects need all the parameters listed by the definition; some do not. In the descriptions that are contained in the Input Output Reference, we try to indicate which parts are optional. Usually, these will be the last fields in the object input or definition.

Using the Input-Output Reference Document

To assist you in using the Input Output Reference document, it is grouped similarly to the IDD file. These groups are described now.

Group -- Simulation Parameters

This group of objects, (including, but not limited to: Building, Timestep in Hour, Inside Convection Algorithm, Outside Convection Algorithm, Shadowing Calculations, Zone Volume Capacitance Multiplier), influence the simulation in various ways.

Group -- Location – Climate – Weather File Access

This group of objects, (including, but not limited to: Location, RunPeriod, DesignDay, GroundTemperatures), describe the ambient conditions for the simulation.

Group -- Surface Construction Elements

This group of objects describes the physical properties and configuration for the building envelope and interior elements. That is, the walls, roofs, floors, windows, doors for the building. Objects here are (including, but not limited to: Materia:Regular, Material:Regular-R, Material:WindowGlass, Material:Air, Material:WindowGas).

Group -- Thermal Zone Description/Geometry

Without thermal zones and surfaces, the building can't be simulated. This group of objects (including, but not limited to: Zone, Surface:HeatTransfer, Surface:HeatTransfer:Sub, Surface:HeatTransfer:InternalMass, Surface:Shading:Detached, Surface:Shading:Attached) describes the thermal zone characteristics as well as the details of each surface to be modeled. Included here are shading surfaces.

Group – Schedules

This group of objects allows the user to influence scheduling of many items (such as occupancy density, lighting, thermostatic controls, occupancy activity). In addition, schedules are used to control shading element density on the building. This group of objects, (ScheduleType, DaySchedule, WeekSchedule, Schedule), define the hierarchical structure of the schedule usage in EnergyPlus. ScheduleType specifies limits that are used in the input of schedules. DaySchedule specifies hourly values. WeekSchedule specifies the names of DaySchedules for the twelve days that EnergyPlus can use. Schedule then specifies the time periods for weekschedules.

Group -- Space Gains

Not all the influence for energy consumption in the building is due to envelope and ambient conditions. This group of objects describes other internal gains that may come into play (including, but not limited to: People, Lights, Various Equipment Types).

Group -- ExteriorEnergyUse Equipment

This group of objects, (including, but not limited to: ExteriorLights, ExteriorFuelEquipment, ExteriorWaterEquipment) will allow reporting/metering of these energy using features without direct implication on the facility energy.

Group – Air Flow

An important characteristic of energy consumption in buildings is the air flow between zones and due to natural ventilation (e.g. open windows). This group of objects, (including, but not limited to: Infiltration, Mixing, COMIS), describes those elements.

Group -- Design Objects

This is a special group of objects that set the stage for using EnergyPlus to auto-size components of the HVAC system.

Group -- Performance Curves

This is a referential group of objects that are used in many other objects to specify performance of components, equipments, etc. CURVE:BIQUADRATIC, CURVE:CUBIC, CURVE:QUADRATIC specify appropriate coefficients for those kinds of curves. More details of what these are used for will be shown under the individual objects using the curves.

HVAC -- Simulation Groups

The HVAC groups (including, but not limited to: Node-Branch Management, Loops, Plant Control, Air Distribution, Plant Equipment, Coils, Air Loop Components, Terminal Unit Components, Radiant Convective Units, System Managers, Controllers) are intertwined. Each will be described in a bit more detail.

Group -- Node-Branch Management

This group of objects, (including, but not limited to: Node List, Branch List, Branch, Connector List, PIPE), describe most of the connection network of the HVAC Loops. Node List is usually optional though is prescribed in a few objects.

Group -- Plant-Condenser Loops

These objects, (including, but not limited to: Plant Loop, Condenser Loop, Heat Recovery Loop), describe the top level areas of the HVAC loops that include equipment components.

Group -- Plant-Condenser Control

This group of objects, (including, but not limited to: Plant Operation Schemes, Condenser Operation Schemes, Load Range Based Operation, Uncontrolled Loop Operation), describe how the plant/condenser will operate under load.

Group -- Plant-Condenser Flow Control

The objects here (Mixer, Splitter) describe how the flow through the plant condenser will happen.

Group -- Air Distribution

This object group describes the air portion of the HVAC system. Objects include Air Primary Loop, Outside Air Mixer, Outside Air System.

Group – System Level Managers

This group includes some high level control objects for the HVAC system (System Availability manager, Set Point Manager).

Group – Controllers

Group -- Zone Equipment

Describes Zone Equipment.

Group -- Zone Controls and Thermostats

This group of objects bridges the gap between thermal zone requirements and HVAC. These items specify what temperatures and other specifics are needed in each zone.

Group -- Air Path

Group -- Convective Air Units

Group -- Air Distribution Units

Group -- Radiant Convective Units

Group -- Plant-Condenser Equipment

Group – Coils

Group – Fans

Group -- Evaporative Coolers

Group – Humidifiers

Group -- Desiccant Dehumidifiers

Group -- Heat Recovery

Group -- Electric Load Center-Generator Specifications

Group -- Photovoltaics

Group -- Fluid Properties

This, like the Performance Curves Group, specify properties that are used in various other objects.

Group – Report

There are two kinds of reports currently available.

Extensive detail is available for most of the input objects using the "Report Variable" object. These can appear at a variety of detail levels in the standard output file (**eplusout.eso**) and the meter output file (**eplusout.mtr**). Using these will be crucial in displaying the results to your users.

There are some canned reports available. These address various details of a one-time nature (such as Conduction Transfer Function calculations or displaying surface vertices). One critical report for interface developers is the "Variable Dictionary" report. With this report, all of the "key" strings to specify report variables in the input files can be determined. Note that not all available report variables will be shown in this report – only those that will be used by the input file that generated the file.

An overall view of reports and report variables is included later in this document.

Standard EnergyPlus Units

EnergyPlus expects information in a single unit system (SI). This requires interface developers to convert user inputs from those preferred by architects and engineers into the standard metric units of EnergyPlus. EnergyPlus will not perform any units conversions and will not have any unit conversion routines.

ASCII with no spaces is used for abbreviations. Note that exponents appear without any indication of exponentiation: i.e., kg/m3 not kg/m^3 or kg/m**3. Also note the use of dashes. We have W/m2-K not W/m2*K or W/(m2*K).

At the end we note the "problem" variables – the inputs that have non-standard units. Inputs using these units will have to be changed and the code checked to see how the quantities are used internally.

Table 1. Standard EnergyPlus Units

Quantity	unit	abbreviation
angular degrees	degree	deg
Length	meter	m
Area	square meter	m2
Volume	cubic meter	m3
Time	seconds	s
Frequency	Hertz	Hz
Temperature	Celsius	C
absolute temperature	Kelvin	K
temperature difference	Kelvin	delK
speed	meters per second	m/s
energy (or work)	Joules	J
power	Watts	W
mass	kilograms	kg
force	Newton	N
mass flow	kilograms per second	kg/s
volume flow	cubic meters per second	m3/s
pressure	Pascals	Pa
pressure difference	Pascals	delPa
specific enthalpy	Joules per kilogram	J/kg
density	kilograms per cubic meter	kg/m3
heat flux	watts per square meter	W/m2
specific heat	-------	J/kg-K
conductivity	-------	W/m-K

diffusivity	-------	m2/s
heat transfer coefficient	-------	W/m2-K
R-value	-------	m2-K/W
heating or cooling capacity	Watts	W
electric potential	volts	V
electric current	Amperes	A
illuminace	lux	lx
luminous flux	lumen	lm
luminous intensity	candelas	cd
luminance	candelas per square meter	cd/m2
vapor diffusivity	m2/s	
viscosity	-------	kg/m-s
porosity	-------	m3/m3
thermal gradient coeff for moisture capacity	-------	kg/kg-K
isothermal moisture capacity	-------	m3/kg

EnergyPlus Reports

Several items are used to specify what will appear in the output file(s). The output is described in the next section of this document.

Reports

Several reports have been installed in EnergyPlus.

```
Report,
  A1 , \field Type_of_Report
       \type choice
       \key Variable Dictionary
       \key Surfaces
       \key Construction
  A2 ; \field Name_of_Report
       \note only applicable to key=Surfaces
       \type choice
       \key DXF
       \key 3DRAW
       \key LINES
```

Surfaces

Surfaces can be reported in several ways:

```
Report, Surfaces, DXF;
```

The above IDF specification will produce a DXF file (**eplusout.dxf**) of the surfaces in the IDF file. Several software programs can render this file into something viewable. For example:

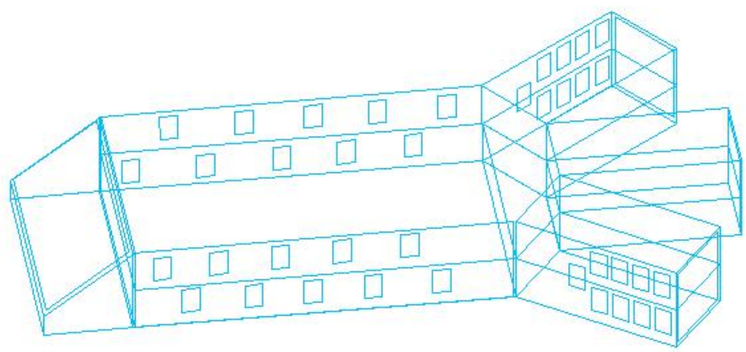

Figure 2. Example building from DXF report.

```
Report, Surfaces, lines;
```

The above IDF line will produce a simple file of line segments that constitute the surfaces in the IDF file.

The following shows an excerpt of "lines" report (**eplusout.sln**) for a single surface. It gives the coordinates in the "standard" EnergyPlus fashion (that is, UpperLeftCorner first and proceeding around, in this case, the four vertices in the surface.

```
     0.00,      0.00,      4.57,      0.00,      0.00,      0.00
     0.00,      0.00,      0.00,     15.24,      0.00,      0.00
    15.24,      0.00,      0.00,     15.24,      0.00,      4.57
    15.24,      0.00,      4.57,      0.00,      0.00,      4.57
```

Variable Dictionary

This report (**eplusout.rdd**) may be necessary before you can ask for specific output variables. Variables available, to some extent, depend on the simulation input. Variables are "set up" during the initial "get input" processing done within the modules. Therefore, an item that is specific to a certain type of coil would not be available if that coil were not used during the simulation. This command will produce a list of variables available for reporting.

```
Report, Variable Dictionary;
```

An example of the results:

```
Program Version,EnergyPlus, 1.0, Beta 1, Build 012
 Var Type,Var Report Type,Variable Name [Units]
Zone,Average,Outdoor Dry Bulb [C]
Zone,Average,Outdoor Barometric Pressure [Pa]
Zone,Average,Outdoor Wet Bulb [C]
Zone,Average,Outdoor Humidity [kgWater/kgAir]
Zone,Average,Wind Speed [m/s]
Zone,Average,Wind Direction [degree]
Zone,Average,Sky Temperature [C]
Zone,Average,Diffuse Solar [W/m2]
Zone,Average,Direct Solar [W/m2]
Zone,Average,Ground Reflected Solar [W/m2]
Zone,Average,Ground Temperature [C]
Zone,Average,Outdoor Dew Point [C]
Zone,Sum,Zone Latent Load[J]
Zone,Sum,Lights Return Air Load[J]
Zone,Sum,BaseBoard Heating[J]
Zone,Sum,Electric Load[J]
Zone,Sum,Gas Equipment Load[J]
Zone,Sum,Infiltration Heat Loss[J]
Zone,Sum,Infiltration Heat Gain[J]
Zone,Average,Mean Air Temperature[C]
Zone,Average,Number of Occupants[]
Zone,Average,Surface Inside Temperature[C]
Zone,Average,Surface Outside Temperature[C]
Zone,Average,Mean Radiant Temperature[C]
HVAC,Sum,Zone/Sys Sensible Heating Energy[J]
HVAC,Sum,Zone/Sys Sensible Cooling Energy[J]
HVAC,Average,Zone/Sys Sensible Heating Rate[W]
HVAC,Average,Zone/Sys Sensible Cooling Rate[W]
HVAC,Average,Zone/Sys Air Temp[C]
HVAC,Average,Zone Air Humidity Ratio[]
HVAC,Sum,HVACManage Iterations
HVAC,Average,System Node Temp[C]
HVAC,Average,System Node MassFlowRate[kg/s]
HVAC,Average,System Node Humidity Ratio
HVAC,Sum,SimAir Iterations
HVAC,Average,Fan Power[W]
HVAC,Average,Fan Delta Temp[C]
HVAC,Sum,Fan Energy[J]
HVAC,Sum,Total Water Coil Energy[J]
HVAC,Sum,Sensible Water Coil Energy[J]
HVAC,Average,Total Water Coil Rate[W]
HVAC,Average,Sensible Water Coil Rate[W]
HVAC,Sum,ZoneAirEq Iterations
HVAC,Average,Plant Loop Demand[W]
HVAC,Average,Unmet Plant Loop Demand[W]
HVAC,Average,Plant Loop Bypass Fraction
HVAC,Sum,Variable Speed Pump Energy[J]
HVAC,Average,Variable Speed Pump Power[W]
HVAC,Average,Chiller Power[W]
HVAC,Average,Chiller Evap Heat Trans Rate[W]
HVAC,Average,Chiller Cond Heat Trans Rate[W]
HVAC,Average,Chiller Cond Water Outlet Temp[C]
HVAC,Average,Chiller Evap Water Outlet Temp[C]
HVAC,Average,Chiller Evap Water mass flow rate[kg/s]
HVAC,Average,Chiller Cond Water mass flow rate[kg/s]
HVAC,Sum,Purchased Chilled Water Energy[J]
HVAC,Average,Purchased Chilled Water Rate[W]
HVAC,Sum,Purchased Hot Water Energy[J]
HVAC,Average,Purchased Hot Water Rate[W]
HVAC,Average,Condenser Loop Demand[W]
HVAC,Average,Unmet Condenser Loop Demand[W]
HVAC,Average,Condenser Loop Bypass Fraction
```

"**Zone**" variables are calculated and can be reported after each Zone/Heat Balance timestep (ref: TimeSteps in Hour command). "**HVAC**" variables are calculated and can be reported with each variable HVAC timestep. "Average" variables will be averaged over the time interval being reported whereas "sum" variables are summed over that time interval.

Constructions

This is one of the reports that will appear in the "**eplusout.eio**" file. It will show the calculated conduction transfer functions for each construction.

```
Report,Constructions;
```

And the example output:

```
! <Construction>,Construction Name,#Layers,#CTFs,Thermal Conductance {w/m2-K},
!OuterThermalAbsorptance,InnerThermalAbsorptance,OuterSolarAbsorptance,InnerSolarAbsorptance,Roughness
! <Material>,Material Name,Thickness {m},Conductivity {w/m-K},Density {kg/m3},
!Specific Heat {J/kg-K},Resistance {m2-K/w}
! <CTF>,Time,Outside,Cross,Inside,Flux (except final one)
 Construction,R13WALL,    1,    1,    0.900,    0.900,    0.750,    0.750,Rough
 Material,R13LAYER,  0.0000,     0.000E+00,      0.000,        0.000,       2.291
 CTF,    1,       0.0000000    ,       0.0000000    ,       0.0000000    ,       0.0000000
 CTF,    0,      0.43649727    ,      0.43649727    ,      0.43649727
```

Report Variable

As shown above in the report variable dictionary, there are many variables available for reporting.

```
Report Variable,
       \note each Report Variable command picks variables to be put onto the standard output file (.eso)
       \note some variables may not be reported for every simulation
  A1 , \field Key_Value
       \note use '*' (without quotes) to apply this variable to all keys
  A2 , \field Variable_Name
  A3 , \field Reporting_Frequency
       \type choice
       \key detailed
       \key timestep
       \key hourly
       \key daily
       \key monthly
       \key runperiod
  A4 ; \field Schedule_Name
```

Each Report Variable object causes a specific number assignment for outputs. For example, you could request separate reporting for the outside temperature:

```
report variable,*,outdoor dry bulb,timestep;
report variable,*,outdoor dry bulb,hourly;
report variable,*,outdoor dry bulb,monthly;
```

And the following would appear in the standard output file:

```
6,2,Environment,Outdoor Dry Bulb [C] !TimeStep
7,2,Environment,Outdoor Dry Bulb [C] !Hourly
8,2,Environment,Outdoor Dry Bulb [C] !Monthly [Value,Min,Day,Hour,Minute,Max,Day,Hour,Minute]
```

Item # 6 will be listed following the TimeStep timestamp for each timestep. Item #7 will be listed following an hourly timestamp. And item #8 will be listed following a monthly timestamp and has additional fields (because it is an "average" variable) that show the minimum and maximum values with identifying times for those minimum and maximum. An excerpt will illustrate:

```
     2,  1, 7,21, 0, 1, 0.00,15.00,Monday – timestep timestamp
6,17.08889
48,21.39851
49,0.0000000E+00
53,0.0000000E+00
60,21.87214
     2,  1, 7,21, 0, 1, 0.00,60.00,Monday – hourly timestamp
7,16.75555
     4,  1, 7 – monthly timestamp
8,22.77037,15.00000,21, 4,60,32.77778,21,14,60
```

To interpret, the first value (#6) is 17.09°C, #7 is 16.76°C (average for the hour), and #8 is 22.77°C, the average for the month with the low (minimum) of 15°C occurring on 7/21 4:60 (or 5:00) and the high (maximum) occurring on 7/21 14:60 (or 15:00).

Field: Key_Value

This alpha field can be used to make a specific reference for reporting. In addition to the generic variable names listed in the Report Variable Dictionary for the input file, variables will also have a key designator (such as Zone name or Surface name). You can reference the standard output file (**eplusout.eso**) to see just how these look.

```
41,2,ZN001:WALL004,Surface Inside Temperature[C]
42,2,ZN001:WALL004,Surface Outside Temperature[C]
43,2,ZN001:WALL004,Surface Int Convection Coeff[W/m2-K]
44,2,ZN001:WALL004,Surface Ext Convection Coeff[W/m2-K]
46,2,ZONE ONE,Mean Radiant Temperature[C]
47,2,ZONE ONE,Zone-Total Latent Gain[J]
51,2,ZONE ONE,Zone-Total Electric Consumption[J]
58,2,ZONE ONE,Zone/Sys Air Temp[C]
```

For example, in the previous block, the key for the surface variables is **ZN001:WALL004** whereas the key for the zone variables is **ZONE ONE** (note that the space is required and significant for this key).

You can have all keys listed in the standard output file by putting a “*” in this field or you can have specific items listed by putting in a key value.

Field: Variable_Name

This alpha field is the variable name (you don’t have to put on the units) that is shown in the Report Variable Dictionary file.

Field: Reporting_Frequency

This field specifies how often the variable will be listed in the output file. “**Detailed**” will list the value each calculation step (i.e. Zone or HVAC). “**TimeStep**” will be the same as “detailed” for Zone valued variables and will be aggregated to the Zone timestep (i.e. TimeStep in Hour value) for HVAC variables. “**Hourly**” will aggregate the value to the hour. “**Daily**” will aggregate to the day (i.e. one value per day). “**Monthly**” will aggregate to the month (i.e. one value per month). “**RunPeriod**” will aggregate to the runperiod specified (each Design Day is a run period as is each runperiod object).

Field: Schedule_Name

The final field is a schedule name. This can be used to limit the number of lines that appear in your output file. For example, a schedule such as “ON PEAK” or “OFF PEAK” could provide a slice of values. Or, a seasonal schedule could be devised.

Other IDF examples:

```
Report Variable, * , Mean Air Temperature, hourly;
Report Variable, * , Mean Radiant Temperature, timestep;
Report Variable, * , Zone/Sys Sensible Heating Energy, hourly;
Report Variable, * , Zone/Sys Sensible Cooling Energy, hourly;
Report Variable, * , Zone/Sys Air Temp, hourly;
```

Output

EnergyPlus produces several output files as shown in the section on “Running EnergyPlus”. This section will discuss the data contained in the “standard” output file (**eplusout.eso**). It, too, has a data dictionary but unlike the input files, the output data dictionary is contained within the output file. Thus, the basic structure of the standard output file is:

```
Data Dictionary Information
End of Data Dictionary
Data
…
Data
End of Data
```

As with the IDF structure, there are rules associated with the interpretation of the standard output data dictionary. These rules are summarized as follows:

- The first item on each line is an integer which represents the “report code”. This “report code” will be listed in the data section where it will also be the first item on each line, identifying the data. Only 2 lines in the output file will not have an integer as the first item (“End of Data Dictionary” and “End of Data” lines).
- The second item on each line is also an integer. This integer corresponds to the number of items left on the dictionary line. Each string consists of a variable name and units in square brackets. Square brackets are required for all strings. If there are no units associated with a particular variable, then there are no characters between the brackets.

Six standard items appear at the start of every EnergyPlus Standard Output File Data Dictionary:

```
Program Version,EnergyPlus, 1.0, Beta 2, Build 017
1,5,Environment Title[],Latitude[degrees],Longitude[degrees],Time Zone[],Elevation[m]
2,6,Day of Simulation[],Month[],Day of Month[],DST Indicator[1=yes 0=no], Hour[], StartMinute[], EndMinute[], DayType
3,3,Cumulative Day of Simulation[],Month[],Day of Month[],DST Indicator[1=yes 0=no],DayType
4,2,Cumulative Days of Simulation[],Month[]
5,1,Cumulative Days of Simulation[]
```

Item 0 is the program version statement.

Item 1 is produced at the beginning of each new “environment” (design day, run period).

Item 2 is produced prior to any variable reported at the timestep or hourly intervals. Hourly intervals will be shown with a start minute of 0.0 and an end minute of 60.0. Timestep intervals will show the appropriate start and end minutes.

Item 3 is produced prior to any variable reported at the daily interval.

Item 4 is produced prior to any variable reported at the monthly interval.

Item 5 is produced prior to any variable reported at the end of the “environment”.

Following these five standard lines will be the variables requested for reporting from the input file (ref. Report Variable). For example:

```
6,2,Environment,Outdoor Dry Bulb [C] !Hourly
21,2,ZONE ONE,Mean Air Temperature[C] !Hourly
22,2,ZONE ONE,Zone-Total Latent Gain[J] !Hourly
26,2,ZONE ONE,Zone-Total Electric Consumption[J] !Hourly
```

This example illustrates the non-consecutive nature of the “report codes”. Internally, EnergyPlus counts each variable that *could* be reported. This is the assigned “report code”. However, the user may not request each possible variable for reporting. Note that, currently, the requested reporting frequency is shown as a comment (!) line in the standard output file.

The data is produced when the actual simulation is performed (after the warmup days). Data output is simpler in format than the data dictionary lines. From the dictionary above:

```
     1,DENVER COLORADO WINTER,  39.75,-104.87,  -7.00,1610.26
     2,  1, 1,21, 0, 1, 0.00,60.00,Monday
6,-17.22222
21,-17.22219
22,0.0000000E+00
26,0.0000000E+00
     2,  1, 1,21, 0, 2, 0.00,60.00,Monday
6,-17.22222
21,-17.22219
22,0.0000000E+00
26,0.0000000E+00
     2,  1, 1,21, 0, 3, 0.00,60.00,Monday
6,-17.22222
21,-17.22219
22,0.0000000E+00
26,0.0000000E+00
…
```

This output file can be easily turned into a form that is read into commonly used spreadsheet programs where it can be further analyzed, graphed, etc.

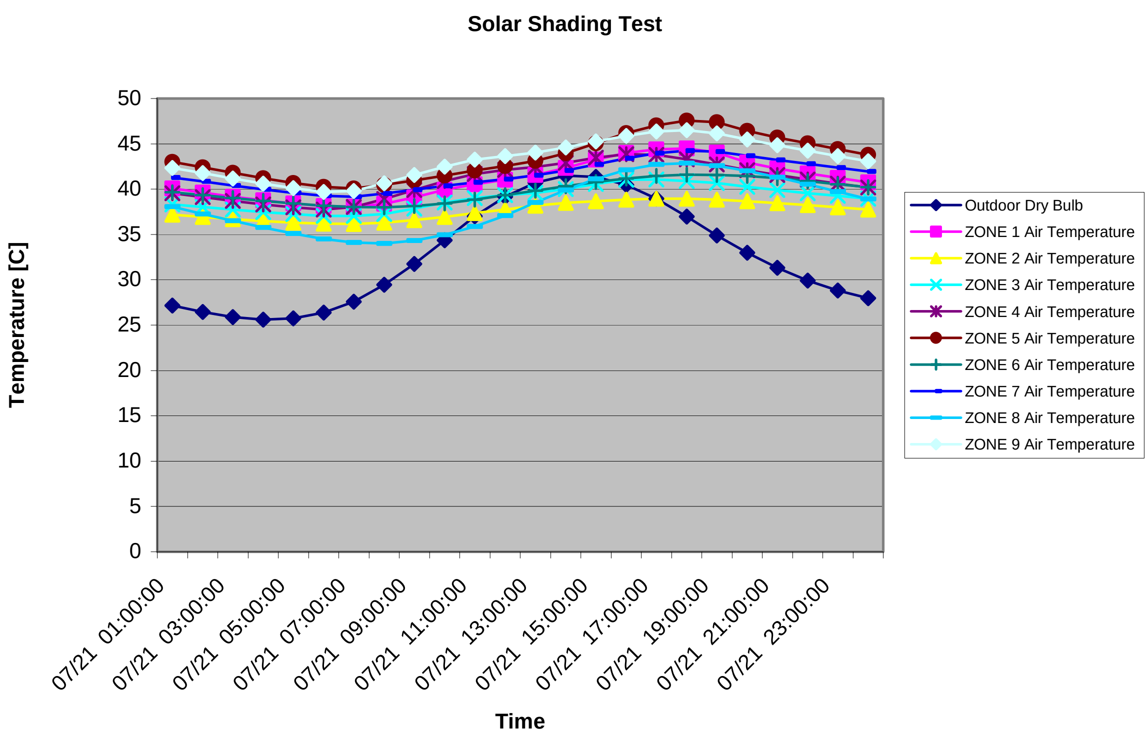

Figure 3. Example Chart from Standard Output File

Weather Data

Weather data in EnergyPlus is a simple text-based format, similar to the input data and output data files. The weather data format includes basic location information in the first eight lines: location (name, state/province/region, country), data source, latitude, longitude, time zone, elevation, peak heating and cooling design conditions, holidays, daylight saving period, typical and extreme periods, two lines for comments, and period covered by the data. The data are also comma-separated and contain much of the same data in the TMY2 weather data set (NREL 1995). EnergyPlus does not require a full year or 8760 (or 8784) hours in its weather files. In fact, EnergyPlus allows and reads subsets of years and even sub-hourly (5 minute, 15 minute) data—the weather format includes a ‘minutes’ field. EnergyPlus comes with a utility that reads standard weather service file types such as TD1440 and DATSAV2 and newer ‘typical year’ weather files such as TMY2 and WYEC2.

The “data dictionary” for EnergyPlus Weather Data is shown below. Note that semi-colons do NOT terminate lines in the EnergyPlus Weather Data.

Note that in the header records where “date” is used, the interpretation is similar to the earlier description.

Table 2. Weather File Date Field Interpretation

Field Contents	Interpretation	Header Applicability
<number>	Julian Day of Year	All date fields
<number> / <number>	Month / Day	All date fields
<number> Month	Day and Month	All date fields
Month <number>	Day and Month	All date fields
<number> Weekday in Month	Numbered weekday of month	Holiday, DaylightSavingPeriod
Last Weekday In Month	Last weekday of month	Holiday, DaylightSavingPeriod

In the table, Month can be one of (January, February, March, April, May, June, July, August, September, October, November, December). Abbreviations of the first three characters are also valid.

In the table, Weekday can be one of (Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday). Abbreviations of the first three characters are also valid.

```
!ESP-r/EnergyPlus Weather Format
!22 November 1999

LOCATION,
   A1, \field city
   \type alpha
   A2, \field State Province Region
   \type alpha
   A3, \field Country
   \type alpha
   A4, \field Source
   \type alpha
   N1, \field WMO
    \type integer
   N2 , \field Latitude
       \units deg
       \minimum -90.0
       \maximum +90.0
       \default 0.0
       \note + is North, - is South,
       \note degree minutes represented in decimal (i.e. 30 minutes is .5)
       \type real
   N3 , \field Longitude
       \units deg
       \minimum -180.0
       \maximum +180.0
       \default 0.0
       \note - is West, + is East,
       \note degree minutes represented in decimal (i.e. 30 minutes is .5)
       \type real
   N4 , \field TimeZone
       \units hr - not on standard units list
       \minimum -12.0
       \maximum +12.0
       \default 0.0
       \note  Time relative to GMT.
       \type real
   N5  \field Elevation
       \units m
       \minimum -1000.0
       \maximum< +9999.9
       \default 0.0
       \type real
```

```
DESIGN CONDITIONS,
     N1, \field Annual Extreme Daily Mean Maximum Dry Bulb Temperature
         \units C
     N2  \field Annual Extreme Daily Mean Minimum Dry Bulb Temperature
         \units C
     N3, \field Annual Extreme Daily Standard Deviation Maximum Dry Bulb Temperature
         \units C
     N4, \field Annual Extreme Daily Standard Deviation Minimum Dry Bulb Temperature
         \units C
     N5  \field 99.6% Heating Dry Bulb Temperature
         \units C
     N6, \field 99% Heating Dry Bulb Temperature
        \units C
     N7, \field 98% Heating Dry Bulb Temperature
        \units C
     N8, \field 0.4% Cooling Dry Bulb Temperature
        \units C
     N9, \field 0.4% Mean Coincident Wet Bulb Temperature
        \units C
     N10, \field 1.0% Cooling Dry Bulb Temperature
        \units C
     N11, \field 1.0% Mean Coincident Wet Bulb Temperature
        \units C
     N12, \field 2.0% Cooling Dry Bulb Temperature
        \units C
     N13, \field 2.0% Mean Coincident Wet Bulb Temperature
        \units C
     N14, \field 0.4% Cooling Dew Point Temperature
        \units C
     N15, \field 0.4% Mean Coincident Dry Bulb Temperature
        \units C
     N16, \field 0.4% Humidity Ratio {?}],
     N17, \field 1.0% Cooling Dew Point Temperature
        \units C
     N18, \field 1.0% Mean Coincident Dry Bulb Temperature
        \units C
     N19, \field 1.0% Humidity Ratio
          \units {?}
     N20, \field 2.0% Cooling Dew Point Temperature
        \units C
     N21, \field 2.0% Mean Coincident Dry Bulb Temperature
        \units C
     N22, \field 2.0% Humidity Ratio
          \units {?}
     N23, \field Daily Range of Dry Bulb Temperature
        \units C
     N23, \field Heating Degree Days Base Temperature
        \units C
     N24, \field Heating Degree Days
     N25, \field Cooling Degree Days Base Temperature
        \units C
     N26 \field Cooling Degree Days
```

```
TYPICAL/EXTREME PERIODS,
     N1, \field Number of Typical/Extreme Periods
     A1, \field Typical/Extreme Period 1
     N2, \field Period 1 Start Date
     N3, \field Period 1 End Date
     A2, \field Typical/Extreme Period 2
     N4, \field Period 2 Start Date
     N5, \field Period 2 End Date
     A3, \field Typical/Extreme Period 3
     N6, \field Period 3 Start Date
     N7, \field Period 3 End Date
     A4, \field Typical/Extreme Period 4
     N8, \field Period 4 Start Date
     N9, \field Period 4 End Date
 -- etc to # of periods entered
```

```
GROUND TEMPERATURES,
     N1, Number of Ground Temperature Depths
     N2, \field Ground Temperature Depth 1
         \units m
       N3, \field Depth 1 Soil Conductivity
         \units W/m-K,
       N4, \field Depth 1 Soil Density
         \units kg/m3
       N5, \field Depth 1 Soil Specific Heat
         \units J/kg-K,
       N6, \field Depth 1 January Average Ground Temperature
        \units C
       N7, \field Depth 1 February Average Ground Temperature
        \units C
       N8, \field Depth 1 March Average Ground Temperature
        \units C
       N9, \field Depth 1 April Average Ground Temperature
        \units C
       N10, \field Depth 1 May Average Ground Temperature
        \units C
       N11, \field Depth 1 June Average Ground Temperature
        \units C
       N12, \field Depth 1 July Average Ground Temperature
        \units C
       N13, \field Depth 1 August Average Ground Temperature
        \units C
       N14, \field Depth 1 September Average Ground Temperature
        \units C
       N15, \field Depth 1 October Average Ground Temperature
        \units C
       N16, \field Depth 1 November Average Ground Temperature
        \units C
       N17, \field Depth 1 December Average Ground Temperature
        \units C
-- etc to # of depths entered
```

```
HOLIDAYS/DAYLIGHT SAVING,
     A1, \field LeapYear Observed
      \type choice
      \key Yes
      \key No
      \note Yes if Leap Year will be observed for this file
      \note No if Leap Year days (29 Feb) should be ignored in this file
     N2, \field Daylight Saving Start Date
     N3, \field Daylight Saving End Date
     A2, \field Holiday 1 Name
     N4, \field Holiday 1 Date
     A3, \field Holiday 2 Name
     N5, \field Holiday 2 Date
     A4, \field Holiday 3 Name
     N6, \field Holiday 3 Date
-- etc to # of Holidays entered
```

```
COMMENTS 1, A1 \field Comments 1]
COMMENTS 2, A1 \field Comments 2]
```

```
DATA PERIODS,
     N1, \field Number of Data Periods
     N2, \field Number of Records per hour
     A1, \field Data Period 1 Name/Description
     A2, \field Data Period 1 Start Day of Week
       \type choice
       \key  Sunday
       \key  Monday
       \key  Tuesday
       \key  Wednesday
       \key  Thursday
       \key  Friday
       \key  Saturday
     N3, \field Data Period 1 Start Date
     N4, \field Data Period 1 End Date
-- etc to # of periods entered
```

```
! Actual data does not have a descriptor
     N1, \field Year
     N2, \field Month
     N3, \field Day
     N4, \field Hour
     N5, \field Minute
     A1, \field Data Source and Uncertainty Flags
     N6, \field Dry Bulb Temperature
        \units C
     N7, \field Dew Point Temperature
        \units C
     N8, \field Relative Humidity
     N9, \field Atmospheric Station Pressure
        \units Pa
     N10, \field Extraterrestrial Horizontal Radiation
        \units Wh/m2
     N11, \field Extraterrestrial Direct Normal Radiation
        \units Wh/m2
     N12, \field Horizontal Infrared Radiation from Sky
        \units Wh/m2
     N13, \field Global Horizontal Radiation
        \units Wh/m2
     N14, \field Direct Normal Radiation
        \units Wh/m2
     N15, \field Diffuse Horizontal Radiation
        \units Wh/m2
     N16, \field Global Horizontal Illuminance
        \units lux
     N17, \field Direct Normal Illuminance
        \units lux
     N18, \field Diffuse Horizontal Illuminance
        \units lux
     N19, \field Zenith Luminance
        \units Cd/m2
     N20, \field Wind Direction
        \units degrees
     N21, \field Wind Speed
        \units m/s
     N22, \field Total Sky Cover
     N23, \field Opaque Sky Cover
     N24, \field Visibility
        \units km
     N25, \field Ceiling Height
        \units m
     N26, \field Present Weather Observation
     N27, \field Present Weather Codes
     N28, \field Precipitable Water
        \units mm
     N29, \field Aerosol Optical Depth
        \units thousandths
     N30, \field Snow Depth
        \units cm
     N31 \field Days Since Last Snowfall
```

Running EnergyPlus

EnergyPlus is written in language conforming to Fortran Standard 90/95. It runs as a 32 bit console (non-Windows) application on Intel compatible computers (Windows NT, Windows 95/98). More explicit details on running EnergyPlus are available in a separate document (Running EnergyPlus in Getting Started). The following files are used to run EnergyPlus:

```
EnergyPlus.exe (the executable file)
Energy+.ini (described below)
Energy+.idd (the input data dictionary file)
In.idf (the input file)
In.epw – optional (weather data file)
```

The input data dictionary and input data file have been discussed in the previous sections of this document.

For weather simulations, EnergyPlus accepts EnergyPlus weather files. Previous versions accepted BLAST formatted weather files and now a BLASTWeatherConverter program is provided. The actual file name is **in.epw**.

The Energy+.ini file is a "standard" Windows™ ini file and can be manipulated using the Windows API calls though EnergyPlus uses standard Fortran to manipulate it. It is a very simple ini file and an example is shown below:

```
[program]
dir=D:\energyplus\
```

The [program] section with the "dir" keyword gives EnergyPlus the directory where to find the Energy+.idd file. This path is used in the "Open" statement. If you leave it blank, EnergyPlus will expect to find it in the directory where the program is running. Energy+.ini and in.idf file should be in the directory from which you are running EnergyPlus.exe.

For the advanced user, there is also the "EPMacro" program, described in the Input Output Reference Document. You run it as a separate program before EnergyPlus (the batch file included in the install and shown in the GettingStarted document contains the commands).

EnergyPlus creates the following files:

Table 3. EnergyPlus Output Files

FileName	Description
Audit.out	Echo of input
Eplusout.err	Error file
Eplusout.eso	Standard Output File
Eplusout.eio	One time output file
Eplusout.rdd	Report Variable Data Dictionary
Eplusout.dxf	DXF (from Report,Surfaces,DXF;)
Eplusout.end	A one line summary of success or failure

The eplusout.err file may contain three levels of errors (Warning, Severe, Fatal) as well as the possibility of just message lines. These errors may be duplicated in other files (such as the standard output file).

Table 4. EnergyPlus Errors

Error Level	Action
Warning	Take note
Severe	Should Fix
Fatal	Program will abort

EnergyPlus produces several messages as it is executing, as a guide to its progress. For example, the run of the 1ZoneUncontrolled input file from Appendix B produces:

```
EnergyPlus Starting
 Warming up
 Initializing Response Factors
 Initializing Window Coefs
 Initializing Solar Calculations
 Warming up
 Warming up
 Performing Simulation
 Warming up
 Initializing Solar Calculations
 Warming up
 Warming up
 Performing Simulation
EnergyPlus Completed Successfully
```

Extensive timing studies and fine-tuning of EnergyPlus is NOT complete. To give you an idea of comparable run times, we present the following (does not include HVAC) with an early version of EnergyPlus running on a 450MHZ machine. Remember, BLAST would be 1 calculation per hour, EnergyPlus (in this case) was 4 calculations per hour.

File	BLAST Per Zone	EnergyPlus Per Zone
GeometryTest (5 Zones, 2 Design Day, Full Weather Year)	13 sec	33 sec
SolarShadingTest (9 Zones, Full Weather Year)	7 sec	25 sec

Table 5. Timings Comparison (EnergyPlus vs. BLAST)

Licensing

In order to make efficient distribution of your interface, you should consider licensing EnergyPlus. First, you must license/register to use EnergyPlus. Since we are distributing EnergyPlus via the World Wide Web, at no charge, it will be easy to get your hands on a copy for testing out your interface. However, you may wish to understand more of the internals and, to make a complete distribution package, will need to at least execute a distribution license.

http://www.energyplus.gov/energyplus_licensing.html contains the details on licensing EnergyPlus for commercial or non-commercial use.

Appendix A. Simple IDF file

```
! 1ZoneUncontrolled.idf
! Basic file description: Basic test for EnergyPlus
! Run:      2 design days.
! Building: Fictional 1 zone building with resistive walls.  No floor.  No Roof.
! Internal: None.
! System:   None.
! Plant:    None.
! SolDis=MinimalShadowing, Aniso, Detailed Interior and Exterior Convection

  Version,
    1.2.0;  !- Version Identifier

  TIMESTEP IN HOUR,
    4;  !- Time Step in Hour

  BUILDING,
    Simple One Zone (no roof-no floor),  !- Building Name
    0.0000000E+00,  !- North Axis {deg}
    Suburbs,  !- Terrain
    3.9999999E-02,  !- Loads Convergence Tolerance Value {W}
    4.0000002E-03,  !- Temperature Convergence Tolerance Value {deltaC}
    MinimalShadowing,  !- Solar Distribution
    25;  !- Maximum Number of Warmup Days

  SOLUTION ALGORITHM,
    CTF;  !- SolutionAlgo

  INSIDE CONVECTION ALGORITHM,
    Detailed;  !- InsideConvectionValue

  OUTSIDE CONVECTION ALGORITHM,
    Detailed;  !- OutsideConvectionValue

  Location,
    DENVER COLORADO,  !- LocationName
    39.750,  !- Latitude {deg}
    -104.870,  !- Longitude {deg}
    -7.0,  !- TimeZone {hr}
    1610.26;  !- Elevation {m}

  DesignDay,
    DENVER COLORADO WINTER,  !- DesignDayName
    -17.22222,  !- Maximum Dry-Bulb Temperature {C}
    0.0000000E+00,  !- Daily Temperature Range {deltaC}
    -17.22222,  !- Humidity Indicating Temperature at Max Temp {C}
    83361.40,  !- Barometric Pressure {Pa}
    4.115816,  !- Wind Speed {m/s}
    169.0000,  !- Wind Direction {deg}
    0.0000000E+00,  !- Sky Clearness
    0,  !- Rain Indicator
    0,  !- Snow Indicator
    21,  !- Day Of Month
    1,  !- Month
    Holiday,  !- Day Type
    0,  !- Daylight Saving Time Indicator
    Wet-Bulb;  !- Humidity Indicating Temperature Type

  DesignDay,
    DENVER COLORADO SUMMER,  !- DesignDayName
    32.77778,  !- Maximum Dry-Bulb Temperature {C}
    17.77778,  !- Daily Temperature Range {deltaC}
    15.00000,  !- Humidity Indicating Temperature at Max Temp {C}
    84058.15,  !- Barometric Pressure {Pa}
    3.971544,  !- Wind Speed {m/s}
    146.0000,  !- Wind Direction {deg}
    1.100000,  !- Sky Clearness
```

```
    0,  !- Rain Indicator
    0,  !- Snow Indicator
    21,  !- Day Of Month
    7,  !- Month
    Monday,  !- Day Type
    1,  !- Daylight Saving Time Indicator
    Wet-Bulb;  !- Humidity Indicating Temperature Type

  MATERIAL:Regular-R,
    R13LAYER,  !- Name
    Rough,  !- Roughness
    2.290965,  !- Thermal Resistance {m2-K/W}
    0.9000000,  !- Absorptance:Thermal
    0.7500000,  !- Absorptance:Solar
    0.7500000;  !- Absorptance:Visible

  CONSTRUCTION,
    R13WALL,  !- Name
    R13LAYER;  !- Outside Layer

  GroundTemperatures,
    18.89,  !- January Ground Temperature {C}
    18.92,  !- February Ground Temperature {C}
    19.02,  !- March Ground Temperature {C}
    19.12,  !- April Ground Temperature {C}
    19.21,  !- May Ground Temperature {C}
    19.23,  !- June Ground Temperature {C}
    19.07,  !- July Ground Temperature {C}
    19.32,  !- August Ground Temperature {C}
    19.09,  !- September Ground Temperature {C}
    19.21,  !- October Ground Temperature {C}
    19.13,  !- November Ground Temperature {C}
    18.96;  !- December Ground Temperature {C}

  ZONE,
    ZONE ONE,  !- Zone Name
    0.0000000E+00,  !- Relative North (to building) {deg}
    0.0000000E+00,  !- X Origin {m}
    0.0000000E+00,  !- Y Origin {m}
    0.0000000E+00,  !- Z Origin {m}
    1,  !- Type
    1,  !- Multiplier
    -100.0000,  !- Ceiling Height {m}
    0.0000000E+00;  !- Volume {m3}

  ScheduleType,
    Fraction,  !- ScheduleType Name
    0.0 : 1.0,  !- range
    CONTINUOUS;  !- Numeric Type

  SurfaceGeometry,
    UpperLeftCorner,  !- SurfaceStartingPosition
    CounterClockWise,  !- VertexEntry
    WorldCoordinateSystem;  !- SurfaceGeometryKey

  Surface:HeatTransfer,
    Zn001:Wall001,  !- User Supplied Surface Name
    Wall,  !- Surface Type
    R13WALL,  !- Construction Name of the Surface
    ZONE ONE,  !- InsideFaceEnvironment
    ExteriorEnvironment,  !- OutsideFaceEnvironment
    ,  !- OutsideFaceEnvironment Object
    SunExposed,  !- Sun Exposure
    WindExposed,  !- Wind Exposure
    0.5000000,  !- View Factor to Ground
    4,  !- Number of Surface Vertex Groups -- Number of (X,Y,Z) groups
    0.0000000E+00,0.0000000E+00,4.572000,  !- X,Y,Z ==> Vertex 1
    0.0000000E+00,0.0000000E+00,0.0000000E+00,  !- X,Y,Z ==> Vertex 2
    15.24000,0.0000000E+00,0.0000000E+00,  !- X,Y,Z ==> Vertex 3
    15.24000,0.0000000E+00,4.572000;  !- X,Y,Z ==> Vertex 4
```

```
Surface:HeatTransfer,
  Zn001:Wall002,  !- User Supplied Surface Name
  Wall,  !- Surface Type
  R13WALL,  !- Construction Name of the Surface
  ZONE ONE,  !- InsideFaceEnvironment
  ExteriorEnvironment,  !- OutsideFaceEnvironment
  ,  !- OutsideFaceEnvironment Object
  SunExposed,  !- Sun Exposure
  WindExposed,  !- Wind Exposure
  0.5000000,  !- View Factor to Ground
  4,  !- Number of Surface Vertex Groups -- Number of (X,Y,Z) groups
  15.24000,0.0000000E+00,4.572000,  !- X,Y,Z ==> Vertex 1
  15.24000,0.0000000E+00,0.0000000E+00,  !- X,Y,Z ==> Vertex 2
  15.24000,15.24000,0.0000000E+00,  !- X,Y,Z ==> Vertex 3
  15.24000,15.24000,4.572000;  !- X,Y,Z ==> Vertex 4

Surface:HeatTransfer,
  Zn001:Wall003,  !- User Supplied Surface Name
  Wall,  !- Surface Type
  R13WALL,  !- Construction Name of the Surface
  ZONE ONE,  !- InsideFaceEnvironment
  ExteriorEnvironment,  !- OutsideFaceEnvironment
  ,  !- OutsideFaceEnvironment Object
  SunExposed,  !- Sun Exposure
  WindExposed,  !- Wind Exposure
  0.5000000,  !- View Factor to Ground
  4,  !- Number of Surface Vertex Groups -- Number of (X,Y,Z) groups
  15.24000,15.24000,4.572000,  !- X,Y,Z ==> Vertex 1
  15.24000,15.24000,0.0000000E+00,  !- X,Y,Z ==> Vertex 2
  0.0000000E+00,15.24000,0.0000000E+00,  !- X,Y,Z ==> Vertex 3
  0.0000000E+00,15.24000,4.572000;  !- X,Y,Z ==> Vertex 4

Surface:HeatTransfer,
  Zn001:Wall004,  !- User Supplied Surface Name
  Wall,  !- Surface Type
  R13WALL,  !- Construction Name of the Surface
  ZONE ONE,  !- InsideFaceEnvironment
  ExteriorEnvironment,  !- OutsideFaceEnvironment
  ,  !- OutsideFaceEnvironment Object
  SunExposed,  !- Sun Exposure
  WindExposed,  !- Wind Exposure
  0.5000000,  !- View Factor to Ground
  4,  !- Number of Surface Vertex Groups -- Number of (X,Y,Z) groups
  0.0000000E+00,15.24000,4.572000,  !- X,Y,Z ==> Vertex 1
  0.0000000E+00,15.24000,0.0000000E+00,  !- X,Y,Z ==> Vertex 2
  0.0000000E+00,0.0000000E+00,0.0000000E+00,  !- X,Y,Z ==> Vertex 3
  0.0000000E+00,0.0000000E+00,4.572000;  !- X,Y,Z ==> Vertex 4

report variable,
  *,  !- Key_Value
  outdoor dry bulb,  !- Variable_Name
  hourly;  !- Reporting_Frequency

report variable,
  *,  !- Key_Value
  Daylight Saving Time Indicator,  !- Variable_Name
  daily;  !- Reporting_Frequency

report variable,
  *,  !- Key_Value
  DayType Index,  !- Variable_Name
  daily;  !- Reporting_Frequency

report variable,
  *,  !- Key_Value
  mean air temperature,  !- Variable_Name
  hourly;  !- Reporting_Frequency

report variable,
  *,  !- Key_Value
  Zone-Total Internal Latent Gain,  !- Variable_Name
```

```
  hourly;  !- Reporting_Frequency

report variable,
  *,  !- Key_Value
  mean radiant temperature,  !- Variable_Name
  hourly;  !- Reporting_Frequency

Report variable,
  *,  !- Key_Value
  Surface Inside Temperature,  !- Variable_Name
  daily;  !- Reporting_Frequency

Report variable,
  *,  !- Key_Value
  Surface Outside Temperature,  !- Variable_Name
  daily;  !- Reporting_Frequency

Report variable,
  *,  !- Key_Value
  Surface Int Convection Coeff,  !- Variable_Name
  daily;  !- Reporting_Frequency

Report variable,
  *,  !- Key_Value
  Surface Ext Convection Coeff,  !- Variable_Name
  daily;  !- Reporting_Frequency

report,
  variable dictionary;  !- Type_of_Report

report,
  surfaces,  !- Type_of_Report
  dxf;  !- Name_of_Report

report,
  construction;  !- Type_of_Report
```